UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/784,498	02/23/2004	Dmitry Grebenev	063170.6658	2208

5073   7590   09/20/2007

BAKER BOTTS L.L.P.
2001 ROSS AVENUE
SUITE 600
DALLAS, TX 75201-2980

EXAMINER
MEHRMANESH, ELMIRA

ART UNIT	PAPER NUMBER
2113	

NOTIFICATION DATE	DELIVERY MODE
09/20/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ptomail1@bakerbotts.com
glenda.orrantia@bakerbotts.com

	Application No.	Applicant(s)
	10/784,498	GREBENEV, DMITRY
Office Action Summary	Examiner	Art Unit
	Elmira Mehrmanesh	2113

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12 July 2007*.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application

6)☐ Other: _____.

DETAILED ACTION

This action is in response to an amendment filed on June 12, 2007 for the

application of Grebenev, for a "Kernel-level method of flagging problems in applications"

filed February 23, 2004.

Claims 1-20 are pending in the present application.

Claims 1-20 are rejected under 35 USC § 102.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-20 are rejected under 35 U.S.C. 102(b) as being anticipated by Chen et

al. (U.S. Patent No. 5,684,945).

As per claim 1, Chen discloses a method of identifying problems in applications

(Fig. 1), comprising:

monitoring (col. 6, lines 61-63 and Fig. 1, element 90, *performance tool*) at a

kernel level system resource usage of one or more running applications (col. 26, lines

31-35 and 49-52) without modifying run-time environments of the running applications

(col. 21, lines 65-69 through col. 22, lines 1-2 and 42-52)

and identifying (col. 26, lines 57-63, *host name, process ID*) from the monitored

system usage (col. 16, lines 19-23, *threshold alarm*) an application whose system

usage pattern satisfies a predetermined criteria (col. 87, lines 53-56) associated with

one or more problems (col. 92, lines 56-60).

As per claim 2, Chen discloses the system resource usage comprises one or

more processes that the one or more running applications have spawned (col. 37, lines

55-63).

As per claim 3, Chen discloses the system resource usage comprises central

processing unit usage of the one or more running applications (col. 8, lines 25-31).

As per claim 4, Chen discloses the system resource usage comprises memory

usage of the one or more running applications (col. 26, lines 49-52).

As per claim 5, Chen discloses producing an output comprising at least the

system resource usage associated with each of the one or more running applications

(col. 9, lines 41-51, *data display system*) and (col. 87, lines 25-28, *output of a data

filter*).

As per claim 6, Chen discloses identifying (col. 26, lines 57-63, *host name,

process ID*) from the output (col. 9, lines 41-51, *data display system*) an application

whose system usage pattern satisfies (col. 16, lines 19-23, *threshold alarm*) a predetermined criteria (col. 87, lines 53-56) associated with one or more problems (col. 87, lines 10-15) and (col. 92, lines 56-60).

As per claim 7, Chen discloses the predetermined criteria is an increase in amount of the system resource usage (col. 92, lines 47-67) from a first period to a second period (col. 87, lines 10-15). Chen discloses the flexibility of defining filters (col. 93, lines 6-10) and observing the change in the statistic values taken at different times (col. 50, lines 46-51).

As per claim 8, Chen discloses the predetermined criteria is a continuous increase in amount of the system resource usage (col. 92, lines 47-67) from a first period to a second period (col. 87, lines 10-15). Chen discloses the flexibility of defining filters (col. 93, lines 6-10) and observing the change in the statistic values taken at different times (col. 50, lines 46-51).

As per claim 9, Chen discloses using an available kernel level tool to obtain data associated with the system resource usage (Fig. 1, element 90).

As per claim 10, Chen discloses using an available kernel level tool to obtain data that includes the system resource usage (Fig. 1, element 90) and filtering the data

to obtain a selected system resource usage (col. 87, lines 10-15).

As per claim 11, Chen discloses using the filtered data to identify (col. 26, lines 57-63, *host name, process ID*) an application (col. 16, lines 19-23, *threshold alarm*) an application whose system usage pattern satisfies a predetermined criteria (col. 87, lines 53-56) associated with one or more problems (col. 92, lines 56-60).

As per claim 12, Chen discloses a method of identifying memory problems in applications (col. 27, lines 47-67 through col. 28, lines 1-14) and (col. 81, lines 38-45), comprising:

monitoring (col. 6, lines 61-63 and Fig. 1, element 90, *performance tool*) at a kernel level memory usage of a running application (col. 26, lines 31-35 and 49-52) without modifying a run-time environment of the running application (col. 21, lines 65-69 through col. 22, lines 1-2 and 42-52)

and producing an output (col. 9, lines 41-51, *data display system*) and (col. 27, lines 47-67 through col. 28, lines 1-14) comprising at least the memory usage (col. 26, lines 49-52).

As per claim 13, Chen discloses analyzing the output (col. 81, lines 38-45) to identify a memory problem (col. 26, lines 49-52).

As per claim 14, Chen discloses a method of identifying memory problems (col. 27, lines 47-67 through col. 28, lines 1-14) and (col. 81, lines 38-45) in applications, comprising:

monitoring (col. 6, lines 61-63 and Fig. 1, element 90, *performance tool*) at a kernel level memory usage of a running application (col. 26, lines 31-35 and 49-52) without modifying a run-time environment of the running application (col. 21, lines 65-69 through col. 22, lines 1-2 and 42-52)

and producing an output (col. 9, lines 41-51, *data display system*) and (col. 27, lines 47-67 through col. 28, lines 1-14) comprising at least the memory usage of one or more running applications (col. 26, lines 49-52)

and identifying (col. 26, lines 57-63, *host name, process ID*) from the output (col. 27, lines 47-67 through col. 28, lines 1-14) an application whose memory usage (col. 26, lines 49-52) pattern (col. 87, lines 53-56) associated with one or more problems (col. 92, lines 56-60).

As per claim 15, Chen discloses a method of identifying memory problems (col. 27, lines 47-67 through col. 28, lines 1-14) and (col. 81, lines 38-45) in applications, comprising:

monitoring (col. 6, lines 61-63 and Fig. 1, element 90, *performance tool*) at a kernel level memory usage of a running application (col. 26, lines 31-35 and 49-52) without modifying a run-time environment of the running application (col. 21, lines 65-69 through col. 22, lines 1-2 and 42-52)

and identifying (col. 26, lines 57-63, *host name, process ID*) from the monitored

memory usage (col. 26, lines 49-52) an application whose memory usage (col. 26, lines

49-52) pattern (col. 87, lines 53-56) associated with one or more problems (col. 92, lines

56-60).

As per claim 16, Chen discloses the monitored memory usage comprises at least

a stack memory, data memory, and text memory (col. 26, lines 49-63).

As per claim 17, Chen discloses a method of identifying memory problems in

applications (col. 27, lines 47-67 through col. 28, lines 1-14) and (col. 81, lines 38-45),

comprising:

collecting (col. 6, lines 61-63 and Fig. 1, element 90, *performance tool*) system

recourse usage at a kernel level of the running applications (col. 26, lines 31-35 and 49-

52) without modifying run-time environments of the running applications (col. 21, lines

65-69 through col. 22, lines 1-2 and 42-52)

and identifying (col. 26, lines 57-63, *host name, process ID*) from the collected

system resource usage (col. 16, lines 19-23, *threshold alarm*) an application whose

system usage pattern satisfies a predetermined criteria (col. 87, lines 53-56) associated

with one or more problems (col. 92, lines 56-60).

As per claim 18, Chen discloses a system for identifying problems in applications

(Fig. 1), comprising:

a data collection module operable to retrieve information about a running

application at a kernel level (col. 6, lines 60-67)

and a data analysis module operable to determine from the retrieved information

an abnormal system usage pattern in the information (col. 81, lines 38-47).

As per claim 19, Chen discloses a program storage device readable by machine,

tangibly embodying a program of instructions executable by the machine to perform

method steps of identifying problems in applications (Fig. 1), comprising:

monitoring (col. 6, lines 61-63 and Fig. 1, element 90, *performance tool*) at a

kernel level system resource usage of one or more running applications (col. 26, lines

31-35 and 49-52) without modifying run-time environments of the running applications

(col. 21, lines 65-69 through col. 22, lines 1-2 and 42-52)

and identifying (col. 26, lines 57-63, *host name, process ID*) from the monitored

system usage (col. 16, lines 19-23, *threshold alarm*) an application whose system

usage pattern satisfies a predetermined criteria (col. 87, lines 53-56) associated with

one or more problems (col. 92, lines 56-60).

As per claim 20, Chen discloses the system resource usage is memory usage

(col. 26, lines 49-63), CPU usage (col. 8, lines 25-31), or one or more spawned

processes (col. 37, lines 55-63), or combinations thereof (col. 8, lines 25-31).

Response to Arguments

Applicant's arguments filed June 12, 2007 have been fully considered but they are not persuasive.

As per claims 1, 12, 14, 15, 17, and 20, in response to applicant's argument that Chen fails to disclose the claimed limitations of "monitoring at a kernel level system resource usage of one or more running applications without modifying run-time environments of the running applications; and identifying from the monitored system usage, an application whose system usage pattern satisfies a predetermined criteria associated with one or more problems." the Examiner respectfully disagrees for the reasons provided below.

Chen discloses "process control", which is an important component of system and network performance tuning is the ability to access and alter the course of a process execution while it is running from any node in the system (col. 93, lines 13-28). The management control tool enables a system administer to monitor/manage the hosts/processes by monitoring the system usage (col. 93, lines 5-10) and customize threshold filters (col. 87, lines 37-41).

Chen discloses, "The output of a data filter can be used to detect out of specification performance and feed that data back into a performance tuning system component to modify and improve the system performance." (col. 87, lines 23-28). Chen further discloses identifying a specific process (*i.e. application*) based on its system usage via the process control routine. Note, col. 94, lines 4-12 wherein Chen discloses the process control routine receives the process snapshot from each node. The

process control routine then sends the process data to the GUI to allow the user to

select and operate on the process data as "objects". The user can select a "sort" button

to reorder the menu of process data by a specific category or process parameter, e.g.,

process ID (PID), process name, process priority, userid of the process owner, **process**

memory utilization, CPU utilization, page faults, etc.

As stated above, using the filters/alarm threshold, and the process control routine

to receive the process data allows a user to sort processes based on a specific category

such as system usage, thus "identifying from the monitored system usage an

application" as recited by claims 1, 12, 14, 15, 17, and 20.

As per claim 18, applicant argues that Chen does not disclose "a data collection

module operable to retrieve information about a running application at a kernel level;

and a data analysis module operable to determine from the retrieved information an

abnormal system usage pattern in the information." The Examiner respectfully

disagrees and would like to point out to col. 90, lines 63-64, wherein Chen discloses

kernel level data collection. Chen further discloses of correlating data patterns from the

received performance data for data analysis (col. 86, lines 66-67 through col. 87, lines

1-15). Chen discloses of a GUI module, which enables the user to receive performance

data and perform performance data analysis (col. 8, lines 11-18). Therefore, Chen's

Data Value receiver subsystem (Fig. 7, element 60) and the data supplier daemon and

a dynamic data supplier of figure 22, in connection with the GUI module, provide

performance data analysis as recited in claim 18.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time

policy as set forth in 37 CFR 1 .136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1 .136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later

than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Elmira Mehrmanesh whose telephone number is (571)

272-5531. The examiner can normally be reached on 8-5 M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Robert W. Beausoliel can be reached on (571) 272-3645. The fax phone

number for the organization where this application or proceeding is assigned is 571-

273-8300.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).